كلية العلوم

القسم : **الرياضيات**

السنة : **الاولى**

المادة : خوارزميات

المحاضرة : الثانية/عملي/

{{ مكتبة A to Z }}

Facebook Group : مكتبة A to Z

كلية العلوم ، كلية الصيدلة ، الهندسة التقنية

2025

4

يمكنكم طلب المحاضرات برسالة نصية (SMS) أو عبر(What's app-Telegram)على الرقم 0931497960

قم بحساب Big-O

① for(int i=0; i<n; i++)
{statments}.
O(n)

② for(int i=0, i<=n; i+=2)
{cout<<i<<" ";}.
(n-k)/m ⟹ (n-0)/2 ⟹ O(n/2).

③ for(int i=1; i<=n; i+=2)
{cout<<i<<" ";}
(n-1)/2 ⟹ $O(\frac{n}{2})$
~~for(int i=1; i<n; i*2)~~

④ for(int i=1; i<n; i*=2)
O(Log2(n))

⑤ for(int i=0; i<n*n*n; i++)
→ {system.out.Println(i);
→ for(int j=2; j<n; j++).
→ system.out.Println(j);}
→ system.Println(n); ⟹ $O(n^4)$.

مثال: زمنية التعقيد

الثابت	التكرار
C_1	n^3+1
C_2	$n^3(1)$
C_3	$n^3(n-1)$
C_4	$n^3(n-2)$
C_5	1

$$T(n) = (n^3+1)C_1 + n^3(C_2) + (n^3(n-1))C_3 + (n^3(n-2))C_4 + C_5$$

$$Tn = \underbrace{(C_1+C_2-C_3-2C_4)}_{a} n^3 + \underbrace{(C_3+C_4)}_{b} n^4 + \underbrace{C_5+C_1}_{d}$$

$$T(n) = bn^4 + an^3 + d$$

```
int sum = 0;
for(int i=1; i<n; i++)
{ for(int j=0; j<2*i; j++)
    sum += j;}
return sum;
```

i = 3	i = 2	i = 1
4,3,2,1,0	5,4,3,2,1,0	2,1,0
7,6,5		
n = 7	n = 5	n = 3

$T(n)$ زمنية التعقيد

الثوابت	التكرار
C_1	1
C_2	$n+1$
C_3	$n(n+2)$
C_4	$n(n+1)$
C_5	1

$$\Rightarrow T(n) = 1+n+1+n(n+2)+n(n+1)+1$$
$$= C_1 + (n+1)C_2 + (n(n+2))C_3 + C_4(n(n+1)) + 5$$
$$= (C_3+C_4)n^2 + (C_1+2C_3+C_4)n + 5$$
$$2an^2 + 4bn + 5$$

TEL:0931497960 - Facebook Group: A to Z مكتبة

```
i=1
while(i<=n)
{j=1;
while(j<=n){
j=j*2;}
i=i+1;}
```

$O(n \log_2 n)$

```
i=1;
while(i<=n)
{j=1;
while(j<=n){
j=j*2;}
i=i+1;}
```

$O(n^2)$

```
switch(k)
case1: for(int i=0;i<n;i++)
       {cout<<i;}
case2: {for(int i=0;i<n;i++)
       {for(int j=0;j<n;j++)
       sum=i+j;}}
default cout<<"error";}
```

$O(n^2)$

TEL:0931497960 - Facebook Group: A to Z مكتبة

```
for(int i=0; i<100; i++)
  {cout<<i;}
  for(int j=0; j<n; j++)
     {cout<<j;}
     O(n).
int cout=0
for(int i=n; i>0; i\=2)
  for(int j=0; j<i; j++)
     cout++;
```

$$\sum_{i=0}^{n} \left(\frac{1}{2}\right)^i 1 = 2n \Rightarrow O(n)$$

البحث الخطي: البحث عن قيمة ضمن المصفوفة باستخدام البحث الخطي.

5	2	7	4	10

key=7

```
if(a[i]==key)
```

// a[0]=5 ? key

// a[1]=2 ? key

نعم a[2]=7 ? key

زمن تعقيد الخوارزمية O(n) بطيئ ولا يتم بالترتيب

TEL:0931497960 - Facebook Group: A to Z مكتبة

```
main()
{int a[20];
 int f=0;
int key,k;
int n;
cin>>n;
for(int i=0;i<n;i++)
    {cin>>a[i];}.

cin>>key;
for(int i=0;i<n;i++)
    {if(a[i]=key)}.
     {f=1; k=i;}

if(f==1)
cout<<"found in"<<k;
else cout<<"not found";
```

14	9	2	5	10

key=5

2	5	9	10	14

mid = (4+0)/2 = 2 → a[mid] = a[2] = 9

key < a[mid] high = 4 → high = mid − 1

TEL:0931497960 - Facebook Group: A to Z مكتبة

mide = $\frac{a+1}{2}$ = 0.5 = 0

a[0] = 2 < key

Low = 1

mid = $\frac{1+1}{2}$ = 1

a[1] = 5 = = key. found.

مقارنة بين البحث الخطي والبحث الثنائي

الثنائي أفضل من الخطي اسرع ويحتاج إلى ترتيب

```
int low=0; o(log2n)
int high=n-1;
int mid;
while (low<=high)
{mid=(low+high)/2;
  if (a[mid]<key)
    {low=mid+1;}
else
if(a[mid]>key)
  {high=mid-1;}
  else
  return mid;
```

انتهت المحاضرة

TEL:0931497960 - Facebook Group: A to Z مكتبة

مكتبة A to Z